## REMARKS

Claims 1-50 are presently pending in this application. Claims 1, 14 and 32-39 were rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent No. 6,269,254 by Mathis (hereinafter "Mathis"). Claims 2-13, 15-31 and 40-50 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Mathis in view of U.S. Patent No. 5,473,680 to Porter.

Claims 1-50 remain in this application.

**Rejection of Claims 1, 14 and 32-39 under § 102(e)**

Applicant has amended independent claims 1, 14, 32, 35 and 37 to include limitations that the pluggable terminal type is created under the direction of an application. The involvement of a runtime application in the creation of a pluggable terminal is supported in Applicant's specification in several ways. In describing pluggable terminals, Applicant explains that pluggable terminals provide "greater flexibility and ease," because "TAPI no longer needs detailed information about how each device works and *users no longer need to program private extensions into the MSP component.*" (Specification, p. 22, lns. 9-18, emphasis added). "[Pluggable terminals allow] third parties to provide their own media processing devices and implement control methods for those devices." (Id., p. 24, lns. 7-8). In other words, pluggable terminals are not part of the original operating system generally, or the original TAPI architecture specifically. They can be created by third parties and "plugged" into the TAPI architecture *after* the operating system and TAPI (and the Media Stream Provider (MSP) component, in particular) have already been loaded. In other words, pluggable terminals can be created by applications, rather than by TAPI or the operating system.

To further demonstrate this point, Applicant elaborates that a pluggable terminal type is created by "wrapping" a media processing device control method. "Wrapping means mapping the data structures and methods of the media processing device control method onto the pluggable terminal type..." (Id., p. 27, lns. 17-18). Applicant discloses a data structure for mapping onto a pluggable terminal type that includes "a company name identifying the company

12

that made the new pluggable terminal." (Id., p. 31, lns 31-32). By including a "company name" as an element of the wrapping data structure, Applicant further teaches that pluggable terminals can be created by third party (i.e., "company")-developed applications, rather than by the operating system.

This ability to use pluggable terminals to allow third party applications to add media processing capabilities to existing MSPs without the need to "revise TAPI APIs or program a private extension into a MSP component" gives extensibility not found in the cited prior art. (Id., p. 22, lns. 7-8). Neither reference cited by the Examiner disclose or even suggest the use of a pluggable terminal type created under the direction of an application, rather than the operating system. Terminal objects in Mathis are not created by third party applications, but rather by methods internal to the JTAPI implementation. For example, Mathis discloses a "provider method" that eventually creates a "terminal connection object". (Mathis, col. 5, lns. 19-32). The types of terminals available in Mathis are fixed (VOICE, DATA and FAX) and there is nothing to suggest that new terminal types could be added in an extensible manner.

Applicant therefore respectfully requests the allowance of independent claims 1, 14, 32, 35 and 37.

Claims 33-34, 36 and 38-39 as now presented include the limitations of their amended independent base claims. Applicant therefore respectfully requests the allowance of these claims since they include limitations not taught or disclosed in the cited references.

### Rejection of Claims 2-13, 15-31 and 40-50 under § 103(a)

As discussed above, Mathis and Porter do not disclose a pluggable terminal type created under the direction of an application. As claims 2-13, 15-31 and 40-50 depend on independent base claims, they incorporate all the limitations of their respective base claims. Because some of those limitations are not found in the cited prior art, Applicants respectfully request the allowance of these dependent claims.

The application is considered in good and proper form for allowance, and the Examiner is respectfully requested to pass this application to issue. If, in the opinion of the Examiner, a

telephone conference would expedite the prosecution of the subject application, the Examiner is invited to call the undersigned attorney.

Respectfully submitted,

John J. Bretscher, Reg. No. 52,651
LEYDIG, VOIT & MAYER, LTD.
Two Prudential Plaza, Suite 4900
180 North Stetson Avenue
Chicago, Illinois 60601-6780
(312) 616-5600 (telephone)
(312) 616-5700 (facsimile)

Date: April 22, 2004